Open Systems Engineering

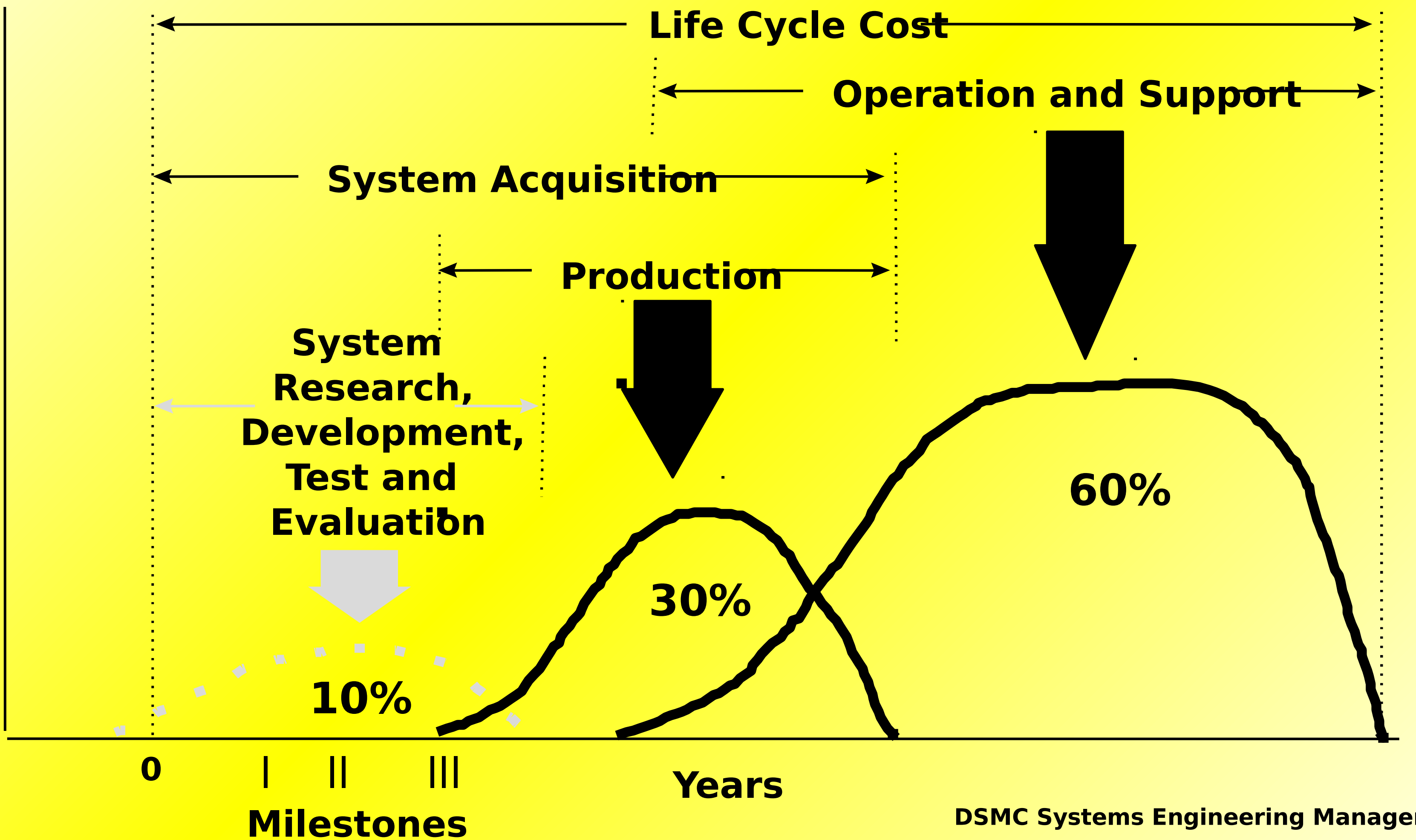
Keeping up with the changing world…

…by designing and building weapon systems using the open system approach.

The New Acquisition Environment

- Unique, Closed Weapons Systems Designs
  - Cost Too Much to Develop
  - Cost Too Much to Support
  - Cost Too Much to Modify
  - Can Not Readily Employ New Technologies
  - Inter-operation Is Less Than Desirable
- Longer Weapon System Life
- Reduced DOD Budget
- Increased Dominance of Commercial Market
- Shortened Technology Cycle Time

Life Cycle Costs



DSMC Systems Engineering Management Guide

What Is an Open System Approach?

The open systems approach is an integrated business and technical strategy to:

- choose commercially supported specifications and standards for selected system interfaces (external, internal, functional, and physical), products, practices, and tools, and
- build systems based on modular hardware and software design.

Business and Technical Approach

It is a business approach to leverage use of commercial products that directs resources to a more intensive preliminary design effort to result in a lifecycle cost reduction. As a business approach it supports the DOD policy initiatives of CAIV, increased competition, and use of commercial products.

It is a technical approach that emphasizes systems engineering, interface control, modular design, and design for upgrade. As a technical approach it supports the engineering goals of design flexibility, risk reduction, configuration control, long term supportability, and enhanced utility.

The Open Systems Approach Makes Sense Whether You are a Manager, Engineer, Logistician, Comptroller, or Contracting Officer

Open Systems Benefits

Relationship to Acquisition Reform

Objectives:

Reduced Cycle Time

Lower Costs

Cost as an Independent Variable

Trade Performance and Schedule for Lower Costs

Performance SPECs

State requirements in terms of needs, not designs

Clear Accountability in Design

Government Controls Performance -- Contractor Designs the Solutions.

Acquisition Reform

Non-Developmental and Commercial Items

Use Existing Technology and Products, If Applicable

OPEN SYSTEMS IS AN ENABLER

Horizontal Technology Insertion

Evolutionary Acquisition

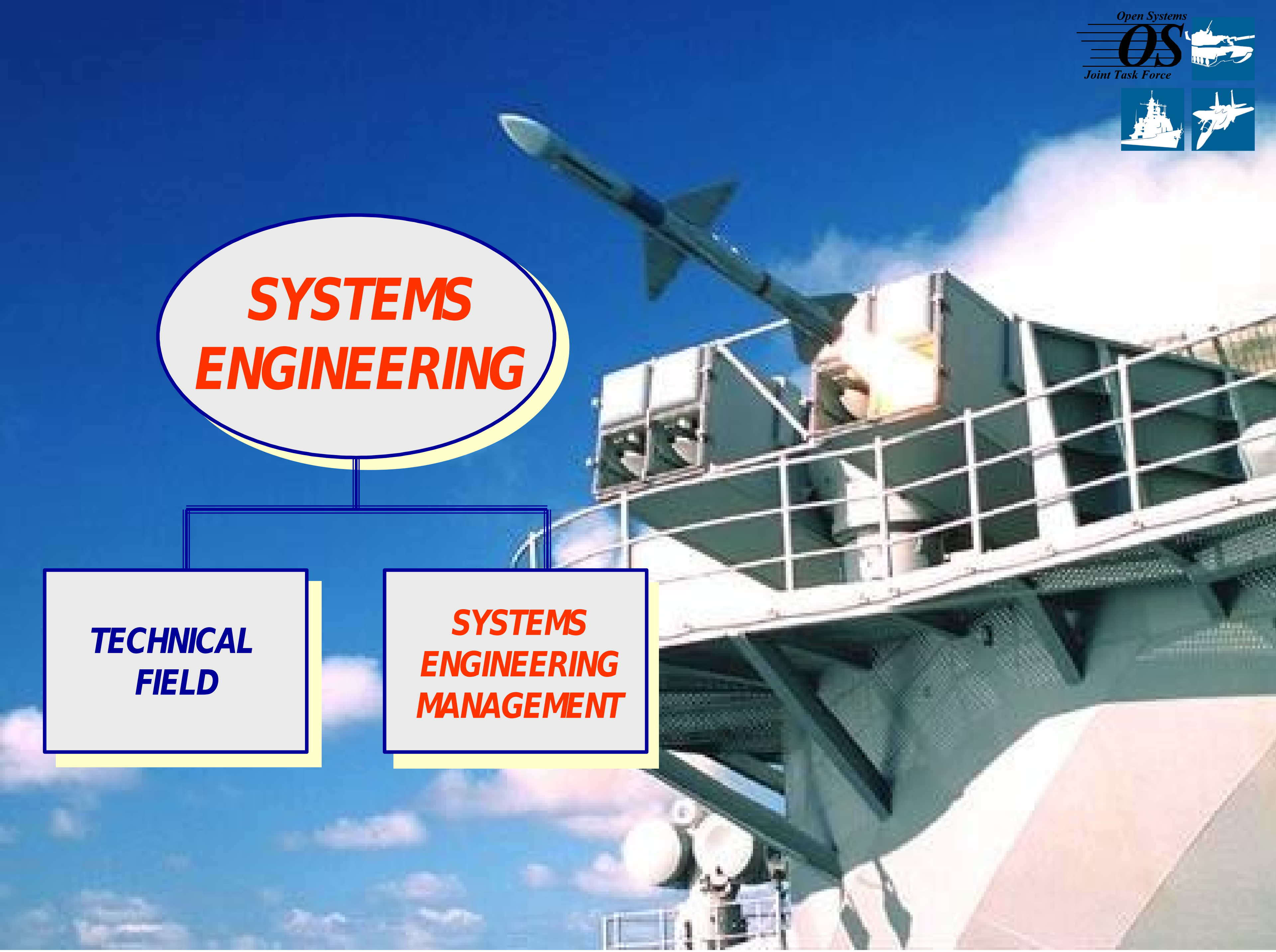
Modernization Through Spares

SUMMARY

- Open System Approach Emphasizes
    - Flexible Interfaces,
    - Maximum Interoperability,
    - Use of Commercial Competitive Products,
    - Enhanced Capacity for Future Upgrade.
- Business and Technical Approach
    - Business Establishes the Need and Availability
    - Technical Supplies the Means
- Associated with Clear Lifecycle Performance, Cost, and Schedule Benefits
- Acquisition Reform Enabler


SYSTEMS ENGINEERING
MANAGEMENT REVIEW

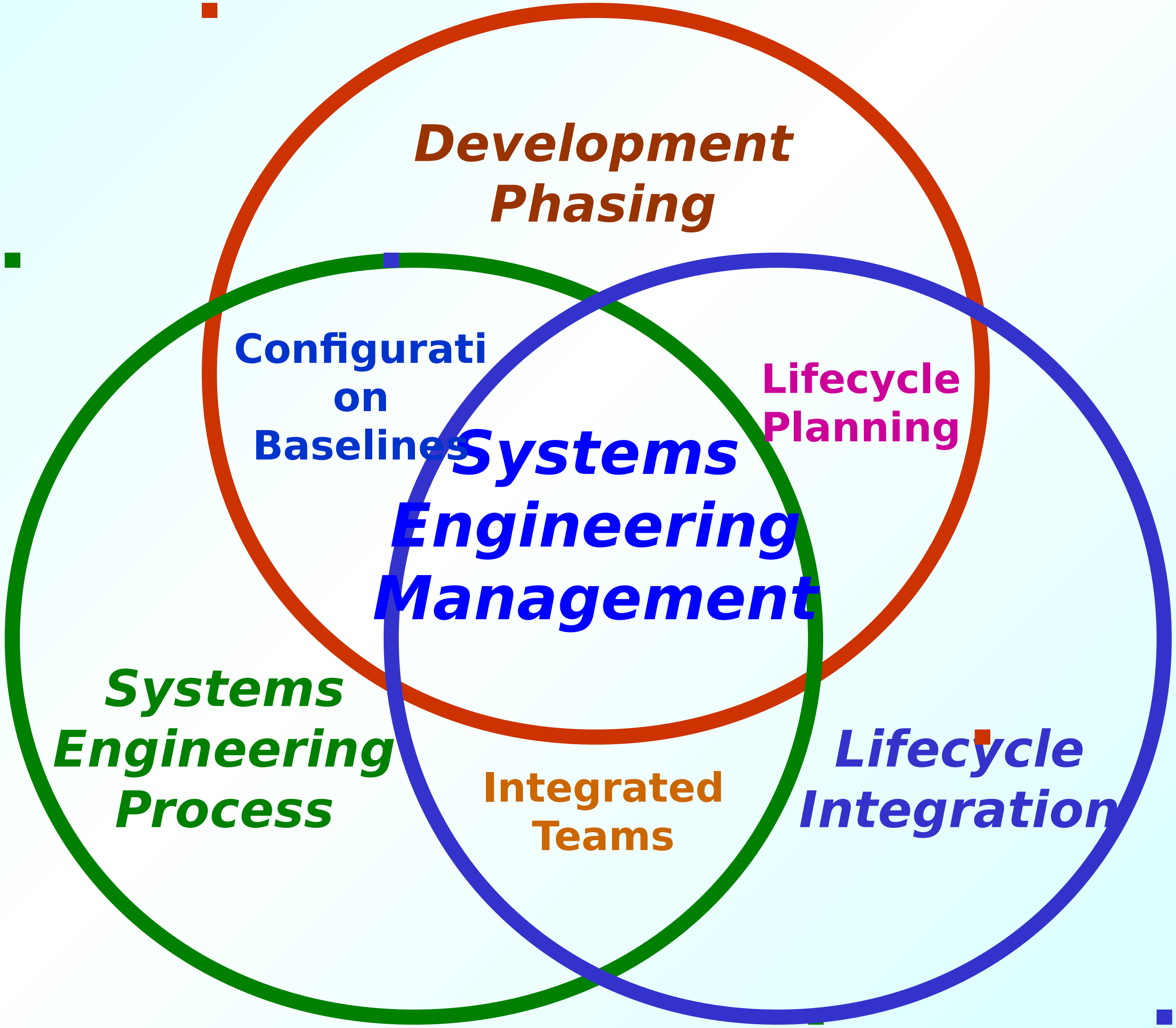
Open Systems
OS
Joint Task Force



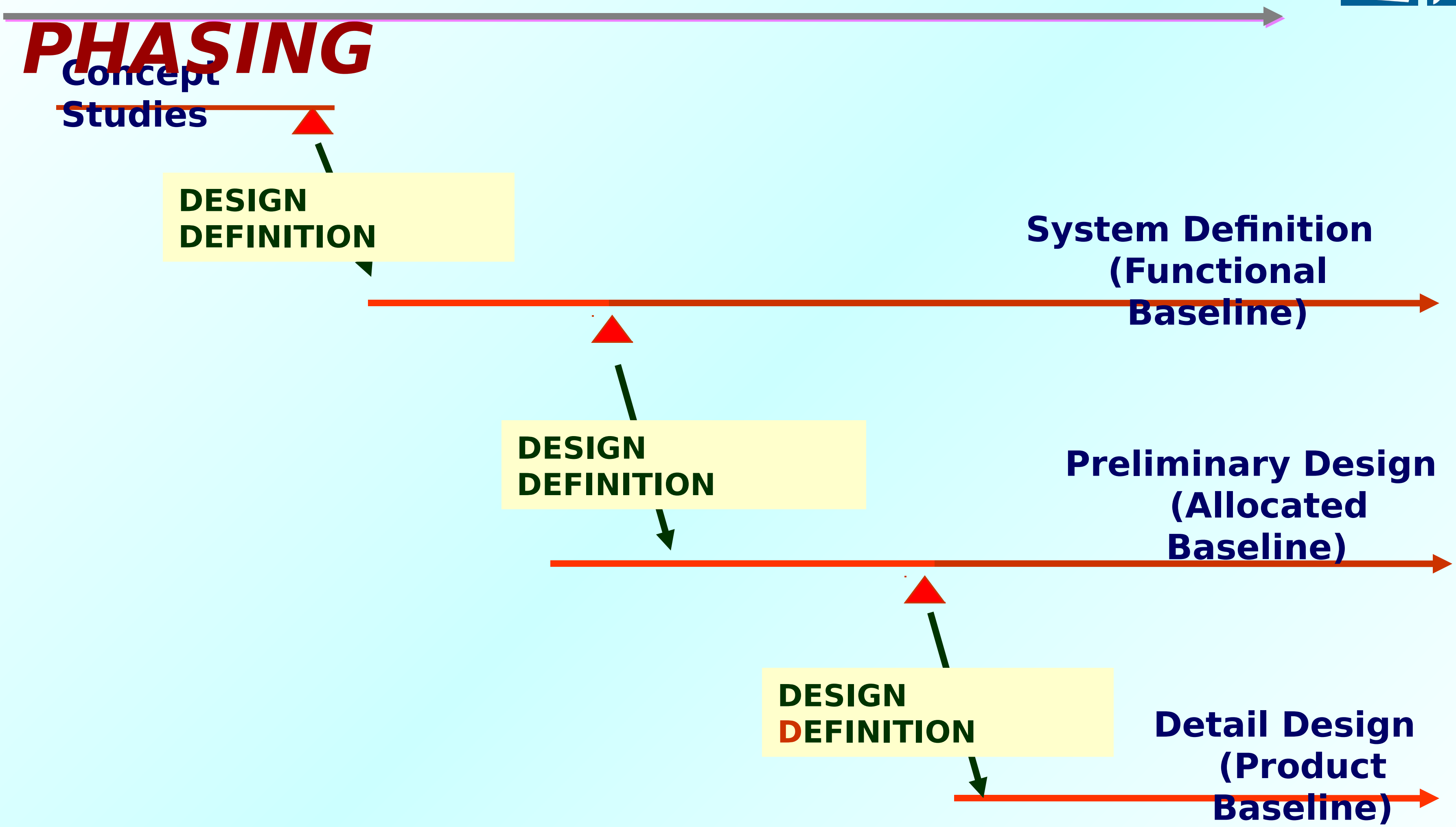
SYSTEMS ENGINEERING

- TECHNICAL FIELD
- SYSTEMS ENGINEERING MANAGEMENT

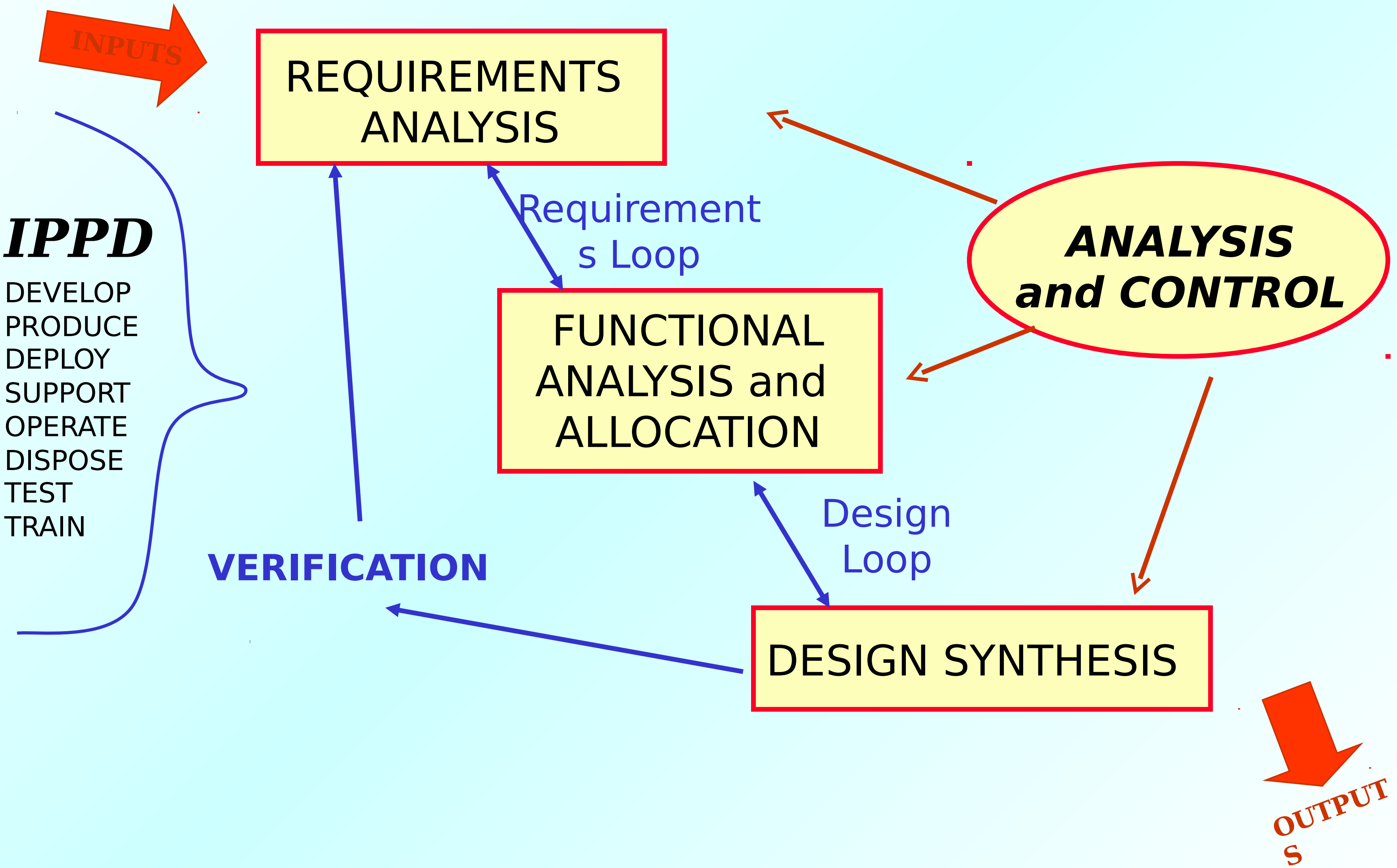
Systems Engineering Management



DEVELOPMENT PHASING



Discipline of the SE Process

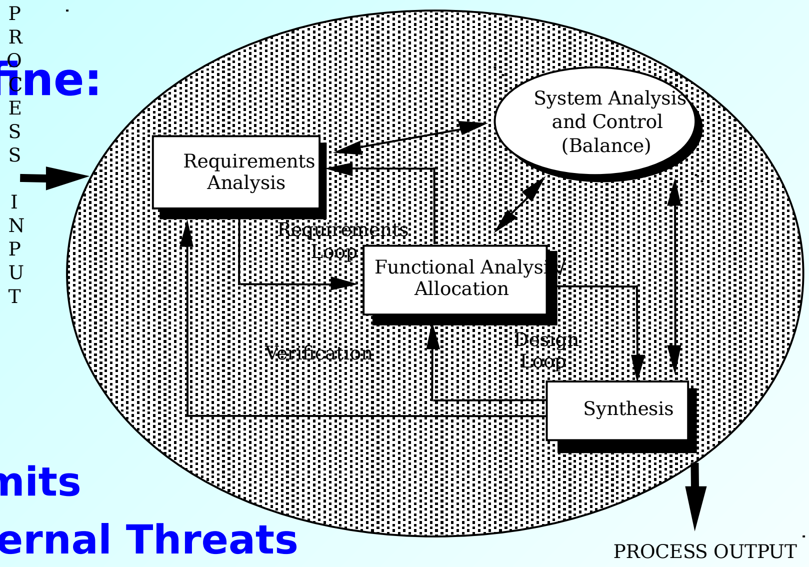


Systems Engineering Process Inputs

- Customer Needs/Objectives/Requirements
  - OP Concept, Missions, and Measures of Effectiveness
  - Environments and Interoperability
  - Constraints (e.g. Cost, Schedule)
  - Technology Base and Engineering Processes.
  - Lifecycle Issues: Producibility, Maintainability, Testability, and Similar
- Output Requirements From Prior Application of SEP.
- Program Decision Requirements.

Requirements Analysis

- **Develop System Functional & Performance Requirements**
  - **Define:**
    - » **What System Must Do**
    - » **How Well It Must Do It**
    - » **Utilization Environment**
    - » **Design Constraints**
- **Performance Requirements Define:**
  - **Quantity - How many**
  - **Quality - How Good**
  - **Coverage - How Far**
  - **Time Lines - When**
    - » **Availability - How Often**
- **Design Constraints Define:**
  - **Environmental Conditions or Limits**
  - **Defense Against Internal or External Threats**
  - **Contract, Customer or Regulatory Standards**



REQUIREMENTS ANALYSIS OUTPUTS THREE VIEWS

- **Operational**
  - Focuses on how the system will be operated by users, including interoperability needs.
  - Establishes HOW WELL and UNDER WHAT CONDITION the system must perform.
- **Functional**
  - Focuses on WHAT system must do to produce required operational behavior.
  - Shows required inputs, outputs, states and transformation rules.
- **Physical**
  - Focuses on HOW the system is constructed.
  - Key to establishing the physical interfaces and technology requirements.

Functional Analysis and Allocation

Allocate Functions

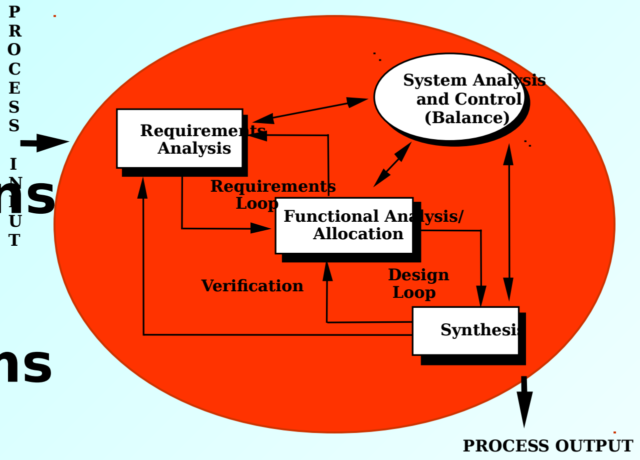
- Decompose Higher Functions

Allocate Performance

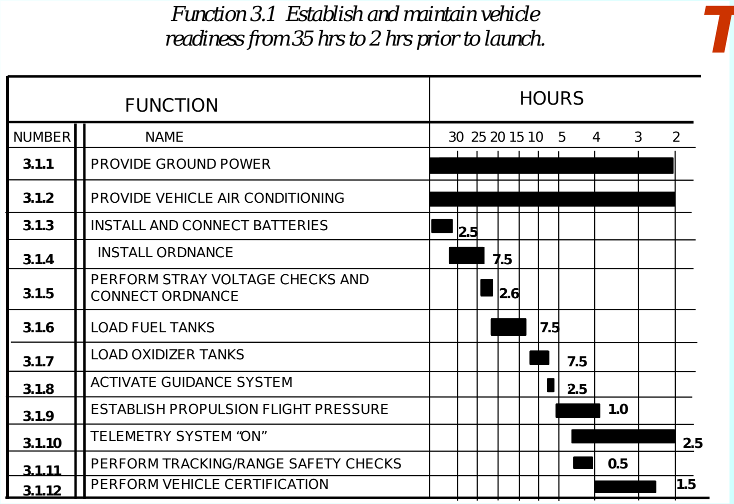
- From Higher to Lower Functions

Functional Descriptions

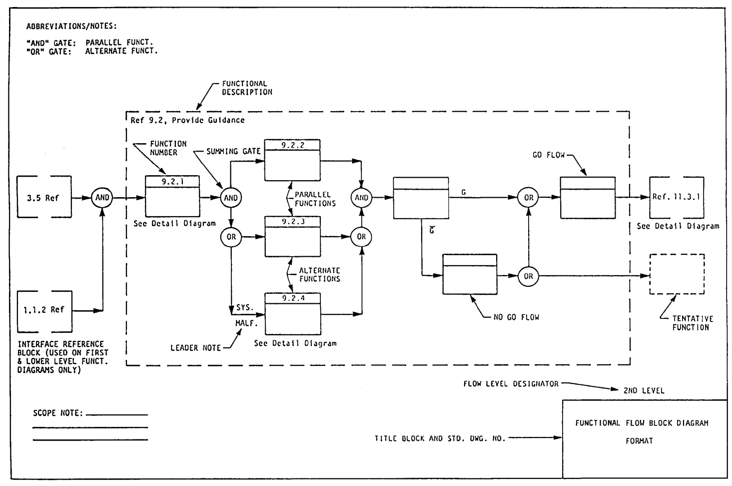
- Functional Flow Block Diagrams
- Time Line Analysis
- Functional Architecture



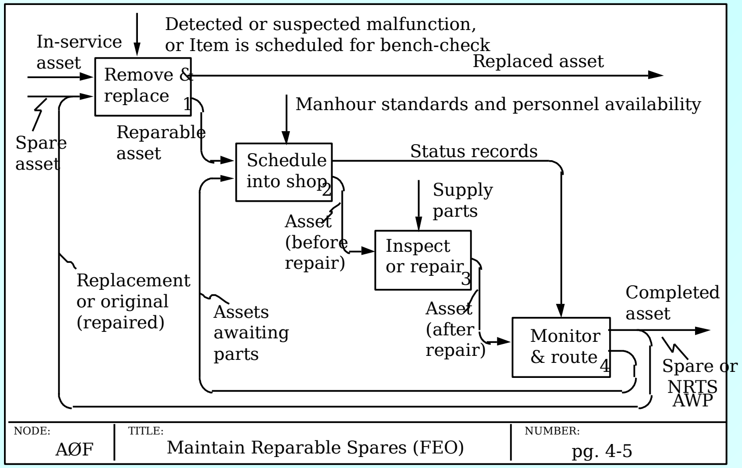
Typical Functional Analysis and Allocation Tools



Time Line Analysis



Functional Flow Block Diagram



IDEF and Similar Functional Interface Tools

Requirements Allocation Sheet	Functional Flow Diagram Title and No. 2.58.4 Provide Guidance Compartment Cooling		Equipment Identification	
Function Name and No.	Functional Performance and Design Requirements	Facility Rqmnts	Nomen clature	CI or Detail Spec No.
2.58.4 Provide Guidance Compartment Cooling	The temperature in the guidance compartment must be maintained at the initial calibration temperature of +0.2 Deg F. The initial calibration temperature of the compartment will be between 66.5 and 68.5 Deg F.			
2.58.4.1 Provide Chilled Coolant (Primary)	A storage capacity for 65 gal of chilled liquid coolant (deionized water) is required. The temperature of the stored coolant must be monitored continuously. The stored coolant must be maintained within a temperature range of 40-50 Deg F. for an indefinite period of time. The coolant supplied must be free of obstructive particles 0.5 micron at all times.			

Requirements Allocation Sheet

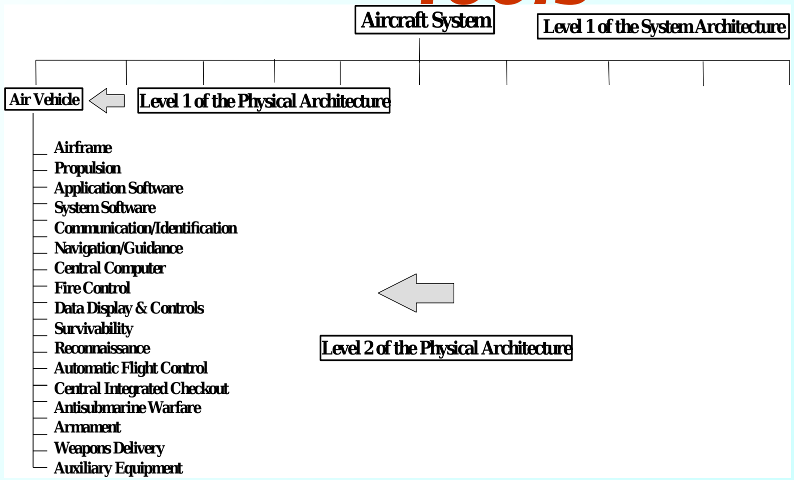
Design Synthesis

- **Outputs:**
  - **Physical Architecture (Product Elements & Software Code).**
  - **Specifications and Baselines.**
  - **Decision Database.**
- **Inputs:**
  - **To Be Transformed: Functional Architecture.**
  - **Enablers: IPTs; Decision Database; Tools: CASE, CAD, CASETS.**
  - **Controls: Constraints, Technical Architectures, GFE, COTS, System Concept & Subsystem Choices; Organizational Procedures.**

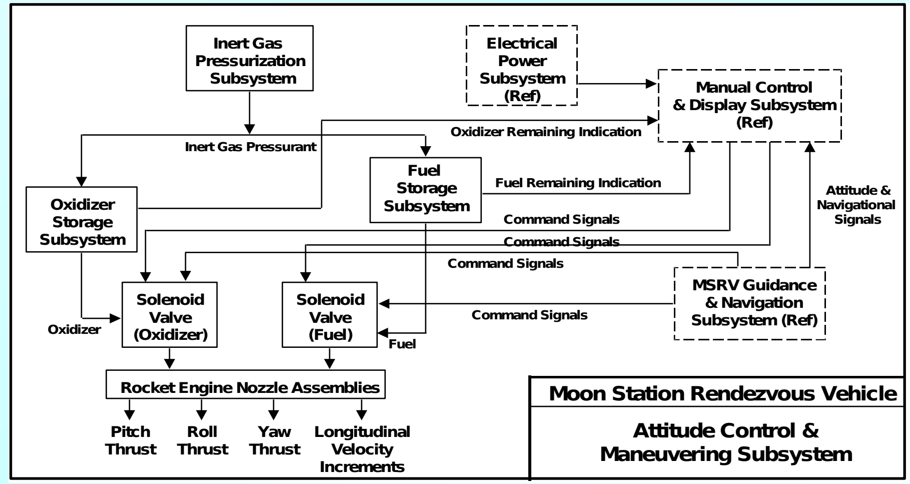
Design Synthesis

- **Activities:**
  - **Allocate Functions & Constraints to System Elements.**
  - **Synthesize System Element Alternatives.**
  - **Assess Technology Alternatives.**
  - **Define Interfaces.**
  - **Define System Product WBS.**
  - **Develop Life Cycle Techniques & Procedures.**
  - **Integrate System Elements.**
  - **Select Preferred Concept/design.**

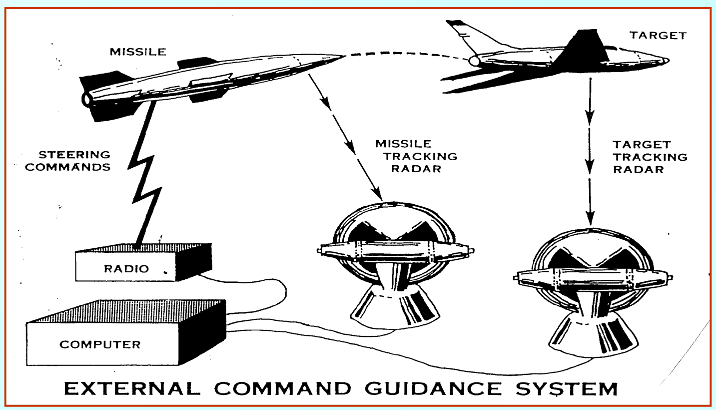
Typical Design Synthesis Tools



WBS



Schematic Block Diagram



Concept Description Sheet

Requirements Allocation Sheet	Functional Flow Diagram Title and No. 2.58.4 Provide Guidance Compartment Cooling		Equipment Identification	
Function Name and No.	Functional Performance and Design Requirements	Facility Rqmnts	Nomenclature	CI or Detail Spec No.
2.58.4 Provide Guidance Compartment Cooling	The temperature in the guidance compartment must be maintained at the initial calibration temperature of +0.2DegF. The initial calibration temperature of the compartment will be between 66.5 and 68.5 Deg F.		Guidance Compartment Cooling System	3.54.5
2.58.4.1 Provide Chilled Coolant (Primary)	A storage capacity for 65 gal of chilled liquid coolant (deionized water) is required. The temperature of the stored coolant must be monitored continuously. The stored coolant must be maintained within a temperature range of 40-50 DegF. for an indefinite period of time. The coolant supplied must be free of obstructive particles 0.5 micron at all times.		Guidance Compartment Coolant Storage Subsystem	3.54.5.1

Requirements Allocation Sheet

Verification

- Each Requirement Must Be Verifiable
  - Specification Section 4 Relates Directly to Section 3
- Confirms That Solution Meets Requirements
- Types of Verification:
  - Inspection
  - Demonstrations
  - Simulations / Analysis
  - Test

SYSTEMS ANALYSIS AND CONTROL

ANALYSIS :

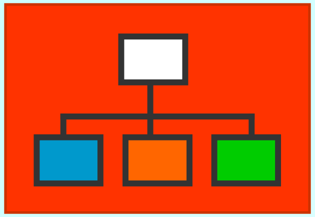
- **Trade Studies**
- **Effectiveness Analysis**
- **QFD**

CONTROL:

- **Work Breakdown Structure**
- **CONFIGURATION AND INTERFACE MANAGEMENT**
- **Data Management**
- **Event Schedules and Tech Reviews**
- **Metrics**
- **Risk Management**
- **Modeling and Simulation**

AND

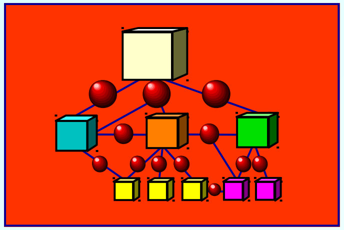
Configuration Management



Management Process for Establishing and Maintaining Consistency of a Product's Performance, Functional, and Physical Attributes With Its Requirements, Design, and Operational Information Throughout Its Life.

Identification, Control, Status Accounting, Audits

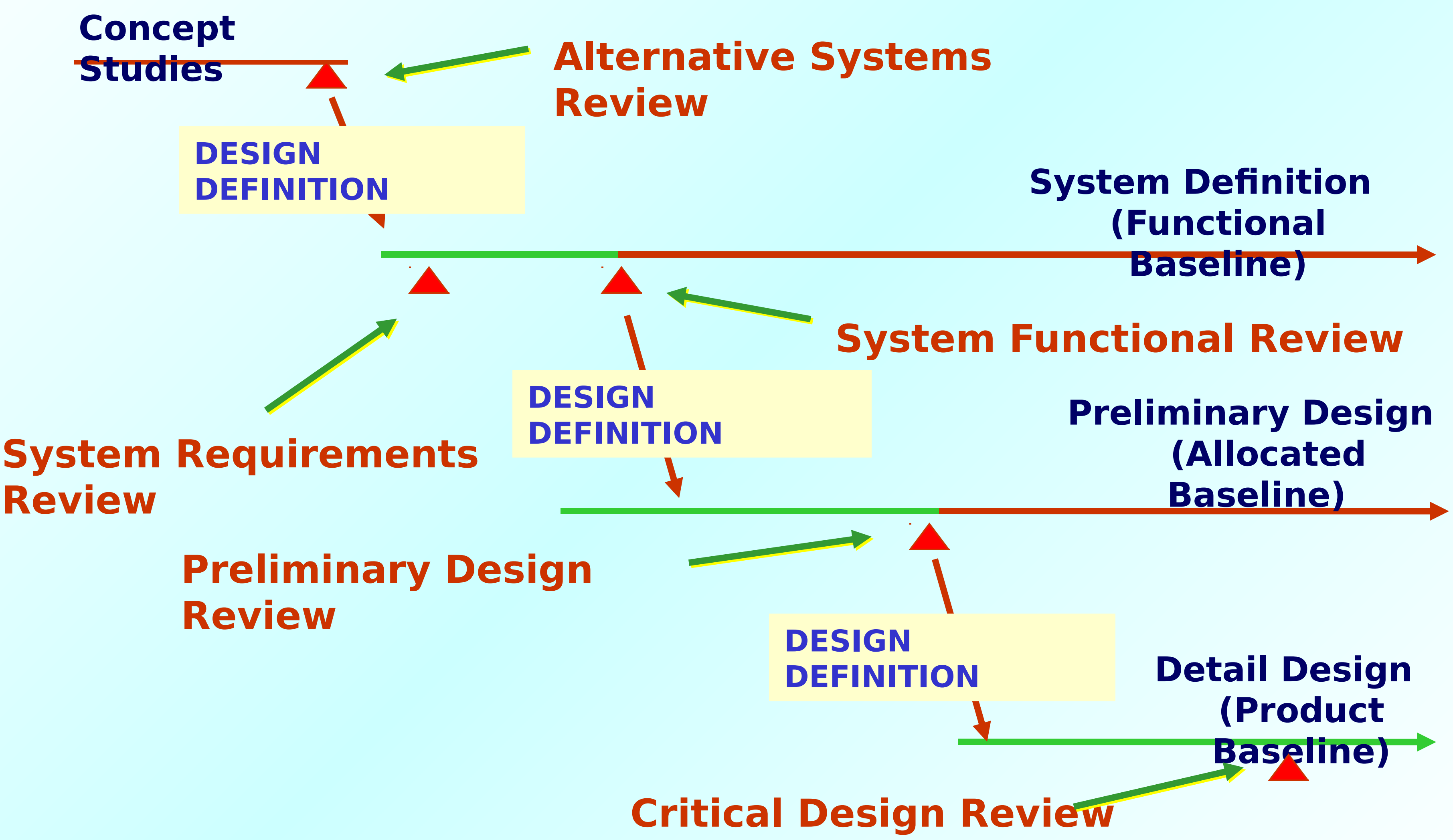
Interface Management



Management Process for Identifying and Controlling Interfaces and the Requirements Associated With Them.

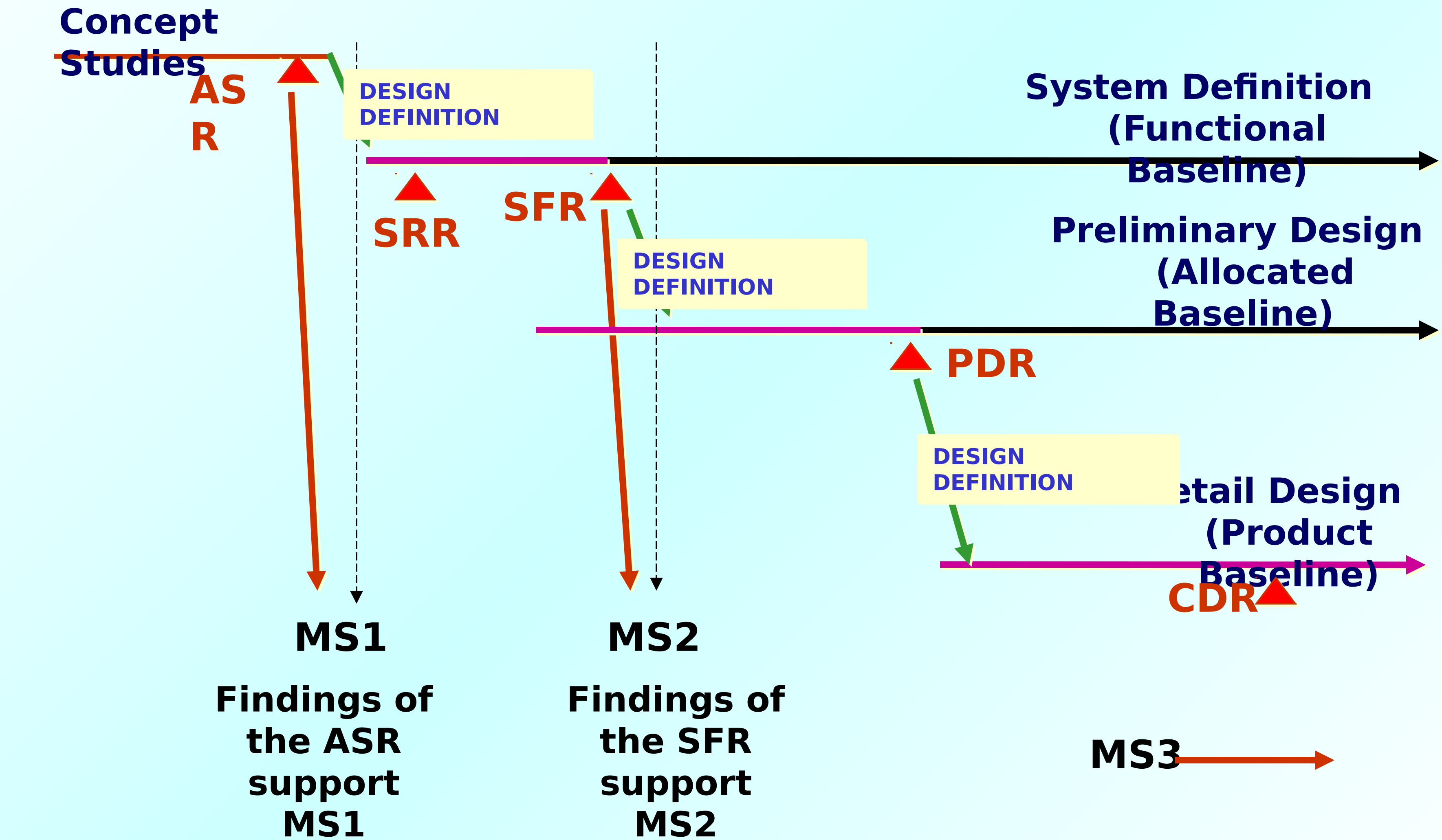
ICWGs, ICD

Technical Reviews



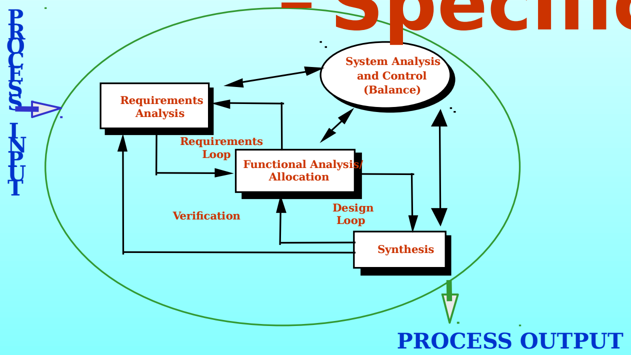
Open Systems
OS
Joint Task Force

Acquisition Milestones



SE PROCESS OUTPUTS

- Level of Development Dependent
  - Decision Data Base
  - System/Configuration Item Architectures
  - Specifications & Baselines



Architectures of a System

Functional

- **Functional requirements**
- **Performance requirements**

Physical

- **Equipment (hardware and software)**

System

- **Products and processes for development, manufacturing, deployment, operations, support, disposal, training, and verification.**

PRIMARY LIFECYCLE FUNCTIONS

8 Primary Functions of Systems Engineering:

- Development
- Production/Construction
- Fielding
- **OPERATION**
- Support
- Disposal
- Training
- Verification

THOSE THAT PERFORM THE PRIMARY FUNCTIONS ARE THE "CUSTOMERS" WHOSE NEEDS FORM THE INPUT TO THE PROCESS, AND THEY ARE THE IPT MEMBERS WHO PERFORM THE SE PROCESS.

Integrated Teaming

Integrates the Lifecycle Functions Into the Design and Development Process

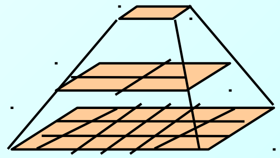
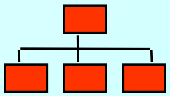
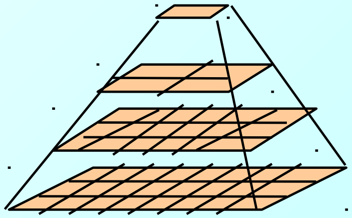
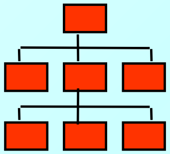
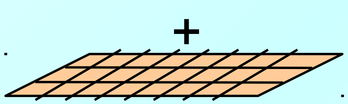
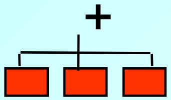
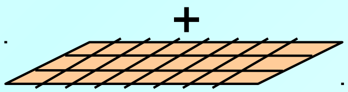
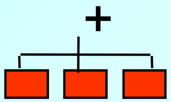
Design-build Teams, Simultaneous Engineering, Concurrent Engineering, IPPD, and Others

IPT Members Perform the SE Process

IPT Members Responsible for Own Functional Planning Based On Team Design

THOSE THAT PERFORM THE PRIMARY FUNCTIONS ARE THE "CUSTOMERS" WHOSE NEEDS FORM THE INPUT TO THE PROCESS, AND THEY ARE THE IPT MEMBERS WHO PERFORM THE SE PROCESS.

SE "Big Picture"

Input	Req'ts Analysis	Functional Analysis & Allocation (Funct Arch)	Synthesis (Phys Arch)	Output (Specs)	WBS (System Arch)	Baseline	Review/ (Phase)
Concept MNS Customer Req'ts	System Level Technical Req'ts			Draft System Spec	Draft Program WBS		ASR (CE)
System ORD#1 Customer Req'ts change? Draft System Spec Phys Arch Mgt Decisions from Tech Reviews	System Level Technical Req'ts			System Spec Draft Performance Specs	Program WBS Draft Contract WBS	Functional	SDR/SFR (PD&RR)
Sub-System ORD#2 Customer Req'ts change? System Spec Draft Perform. Specs Mgt Decisions from Tech Reviews at CI level	System Level Technical Req'ts			System Spec Performance Specs Draft Detail Specs	Program WBS Contract WBS Draft Extensions	Allocated	PDR/CDR FCA (EMD)
Component ORD#3 Customer Req'ts Change? System Spec Performance Specs Draft Detail Specs Mgt Decisions	System Level Technical Req'ts			System, Performance & Detail Specs (TDP)	Program WBS Contract WBS Extensions	Product	PCA (EMD/ PF/D&OS)

Summary

Systems Engineering Management Has Three Elements: Baselines, SEP, and Lifecycle Integration.

There Are Three Baselines: Functional, Allocated, and Product.

The SEP Has Three Main Steps: Requirements Analysis, Functional Analysis and Allocation, and Design Synthesis.

Verification Confirms That Solution Meets Requirements.

The SEP Steps Are Supported by System Analysis and Control Tools. Two Important Ones Are Configuration and Interface Management.

There Are 8 Lifecycle Functions: Design, Production,

OPEN SYSTEMS ENGINEERING MANAGEMENT OVERVIEW

Open Systems

Open Systems Implement Common Interfaces, Services, and Supporting Formats

Open System

- A Collection of Interacting Components Designed to Satisfy Stated Needs With Interface Specifications:
  - Fully Defined
  - Available to the Public
  - Maintained According to Group Consensus
- In Which the Interactions of Components Depend on the Interface Specification and the Implementations of Components Are Conformant to the Specification.

An Open Systems Approach ...

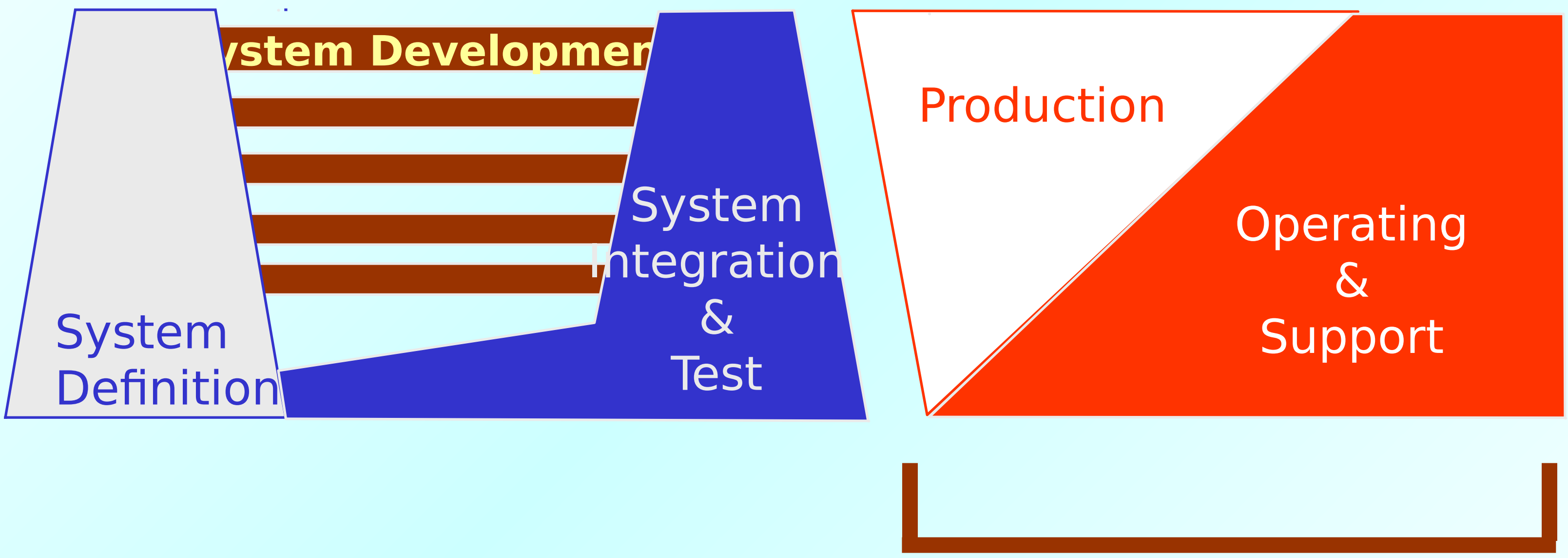
- Is an Integrated Technical and Business Strategy,
  - Uses Modular Hardware and Software Design,
  - Applies Commercial, Widely Used Interface Standards,
  - To Buy, Rather Than Build.

Open Systems is a Systems Engineering Approach

"Traditional" Approach	Open System Approach
Resulted in Unique Interfaces Develop Components *Identify Interfaces*	Adopt Standard Interfaces Acquire Components *Select Interfaces*
Top Down	**Balance Top Down with Bottom Up**

Existing Engineering Methods Emphasize Known User Requirements and Accept an Assumption of Stable Design Through Production and O&S

Current Approach:



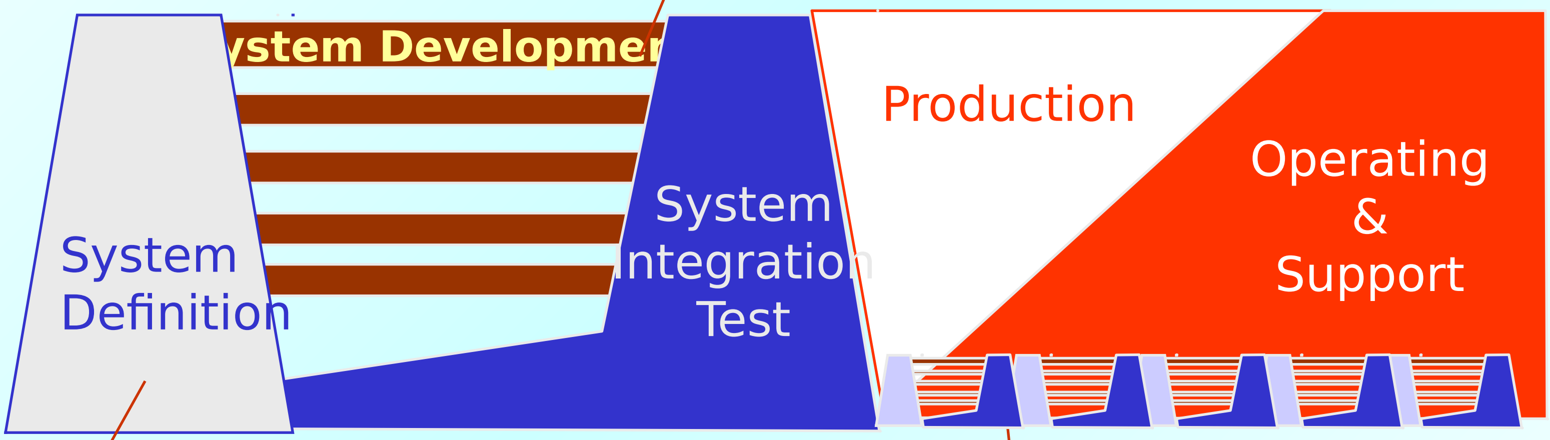
90% of Lifecycle Cost*

***DSMC Systems Engineering Management Guide**

New Approach Must Plan for Rapid Technology Turnover

OS Engineering Model:

Develop Open System Framework & Initial System Configuration



Define System in a manner that accommodates evolution throughout its lifetime

Evolve the System with Available Technologies

From Tutorial on "Developing Military Avionics Systems Using Open Systems Architectures," Chuck Roark, 1997

AN OPEN SYSTEM IS CHARACTERIZED BY

Modular (Compartmentalized) Design.

Multiple Design Solutions Within the Modules With Preference for Competitive Commercial Components.

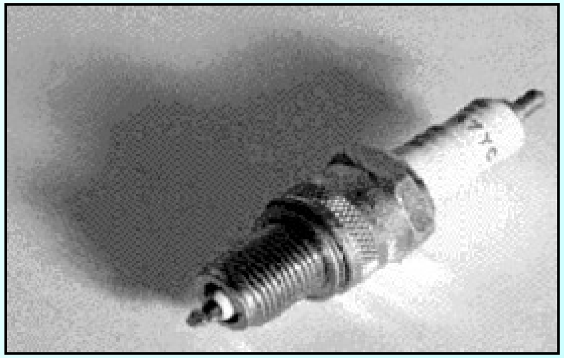
Clear Preference for Open Standards to Define Interfaces.

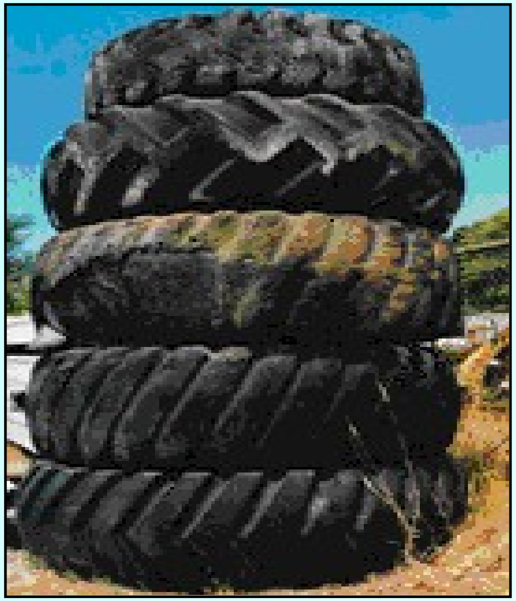
Explicit Provision for Expansion or Upgrading By Component Replacement With Minimal Impact on the System.

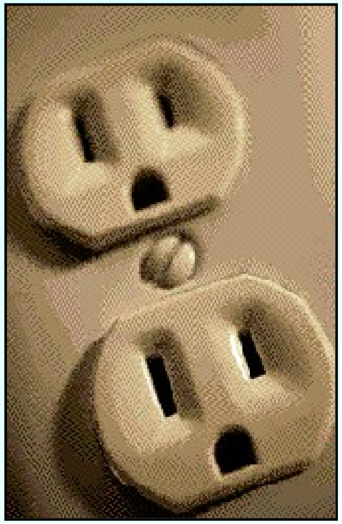
Preferred Characteristics

- **Well Defined, Widely Used, Non-proprietary Interfaces/protocols**
- **Use Of Standards Developed/adopted By Industrially Recognized Standards Bodies**

Typical Open Interface Examples







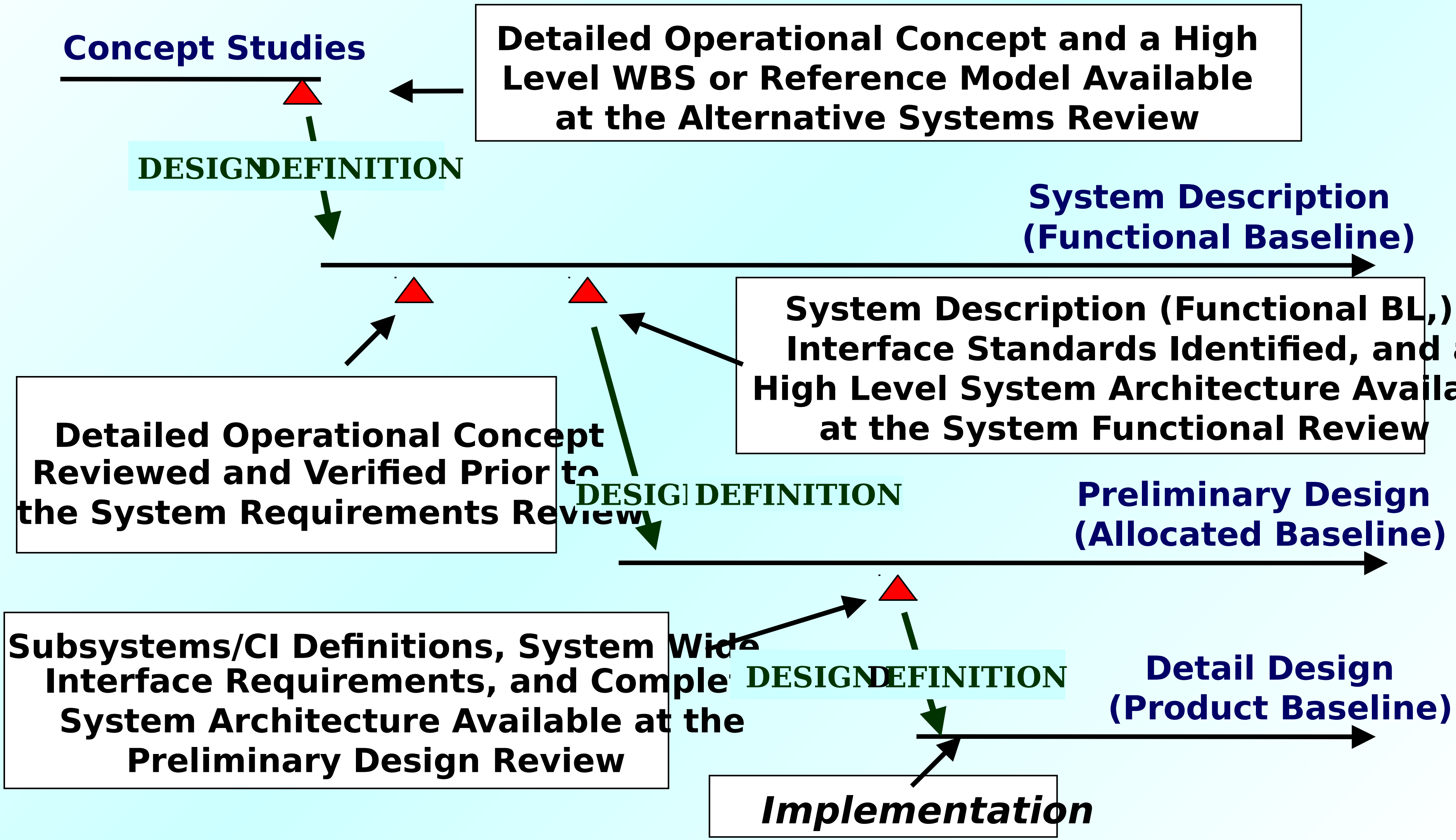
Open Systems Engineering Management

- **Focuses on Design Flexibility to Support Sustainment, Evolution, Upgrade.**
- **Interface *SELECTION* and Control to Enhance Life Cycle Support To Permit Evolution With Technology.**
- **Design anticipates "Change" (Upgrade) Over Time.**
- **Employing Modularity, Based on Well Defined Interfaces, to Isolate Components Likely to Change Over Time.**
- **Multiple Design Solutions Within the Module.**
- **Interface Management Is Key!**

Attributes of an Open System

- **Multiple Sources of Supply**
  - Acquire building blocks from several sources on continuing basis
  - DOD is one of many customers for these building blocks
- **Technology Indifference**
  - Technology Refresh (part requalification)
  - Minimal impact to configuration management
- **Lowers cost of ownership**
- **Better Performing Systems (Technology Insertion)**
- **Greater Intra-operability**

Systems Engineering Management of a Open System Approach



Operational Concept

More Than an Operational Requirements Document (ORD)

Key Features:

Performance Required (ORD)

How the System Will Be Used - Scenarios and Simulations

Dynamic Interfaces: Interoperability, Information Exchange, Log Support, and Other Lifecycle Support

Potential for Alternate Use

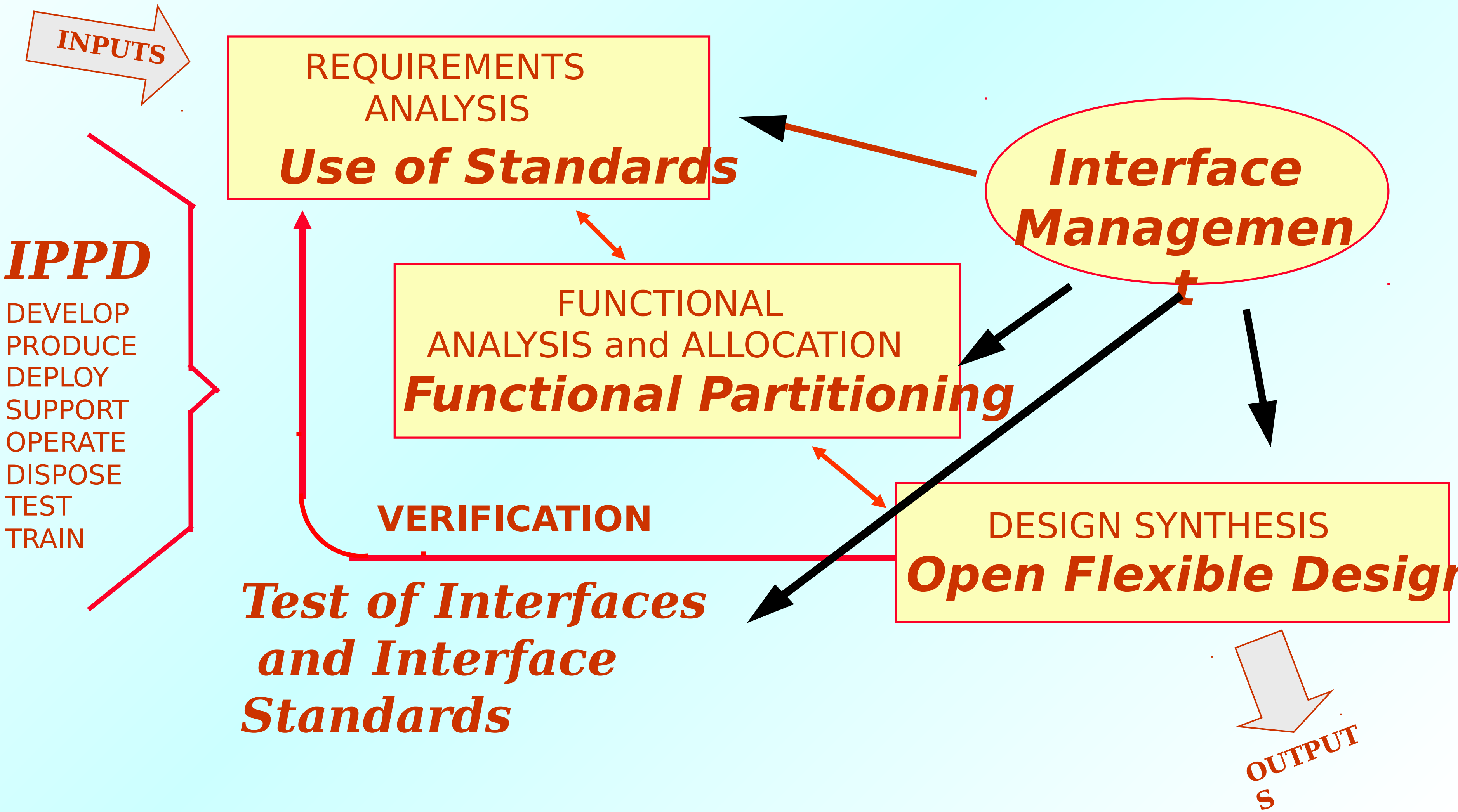
Most Likely Element to Cause The Most Trouble

Operational Architectures Document the Operational Concept

MAJOR ISSUES RESOLVED PRIOR TO SRR!

Designing Open Systems

Demands the Discipline of the SE Proce



Open Systems Design Principles / 1

- **Identify Critical Interfaces to Subsystems or Components Likely To:**
  - **Change Due to Their Dependence on Rapidly Evolving Technology**
  - **Have Increasing Requirements**
  - **Have High Replacement Frequency or Have High Costs**
  - **Have the Highest Obsolescence Risks and the Greatest Opportunity for Future Technology Insertion.**

INTERFACE MANAGEMENT

- **Use Open Standards for These Critical Interfaces That Are:**
  - **Supported by the Broader Community**
  - **Are Considerate of Life-cycle Support Requirements**
  - **Permit Evolution With Advances in Technology and Support Technology Insertion**

INTERFACE MANAGEMENT

Open Systems Design Principles / 2

- **Verify All Performance Requirements and Re-evaluate Their Stringency.**
  - **Reallocate Requirements As Necessary to Permit the Wider Use of Open Standards Throughout the System.**
- **Identify the Lowest Level the Government Maintains Control Over the Interface Standard**
  - **Anticipate How This Level May Change Over Time**
  - **Below This Level, the Contractor Is Permitted to Use Its Best, Perhaps Proprietary, Practices to Improve or Discriminate Its Product in the Marketplace**

REQUIREMENT SANALYSIS

CONFIGURATION MANAGEMENT

Open Systems Design Principles / 3

- **Use a Modular Design Approach Combined With Well-defined Standards-based Interfaces**
  - To Isolate the Effects of Change in Evolving Systems
  - To Reduce the Need for Redesign As the System Is Upgraded

FUNCTIONAL ANALYSIS DESIGN SYNTHESIS

- **Implement Consistent Conformance Management Practices to Ensure That Products Conform to the Established Profile**
  - Prevent Being Limited to One Supplier Who Might Unilaterally Extend That Interface.

CONFORMANCE MANAGEMENT

Requirements Analysis

Review of the Previous SEP Output:

Architectures - Operational, Technical, Functional, Physical, Systems

Interface Definition and Configuration Baselines, Including Specs

Complete, Comprehensive, Concise, & Correct

Consideration of Requirements Change to Enhance OS

Requirements Analysis

Review of New or Revised Requirements:

- ***ORD Revision***
- ***Lifecycle Policy or Procedural Change***
- ***Management Direction***
- ***DOD, Service, or Agency Policy Change***

Analysis of Market Survey

Functional Analysis/Allocation and Design Synthesis

Group Functions or Components Using:

Functional Partitioning

Modular Design

Interface Management

Robust, Flexible, and Minimal or Optimized Interfaces

Verification and Conformance

Conformance Testing

Testing Standards

Testing Components

Operational Testing

Commercial Availability

Balancing Operational Requirement

Every Requirement Must Be Verifiable!

System Analysis and Control

Interface, Configuration and Conformance Management

Configuration Management Is a Formal Process for Controlling and Tracking the System and Component Descriptions. It Is Essential for Assuring the Components Procured Will Accurately Represent the Design.

Interface Management Is the Process of Identifying, Prioritizing, Defining, and Tracking External and Internal Interfaces.

Conformance Management Is the Process That Tracks and Maintains the Interface Requirements Throughout the Lifecycle, and Assures That the Product Meets Those Requirements.
Conformance Management Relates Interface Management to Configuration Management.

System Analysis and Control: Interface Management

Selection/Identification

Specification/Documentation

Interface Control Teams

“Enhanced” Interface Management

External Interfaces Defined in Detail (Operational Architecture) and Put Under Control

All Interfaces Identified and Characterized

Critical Internal Interfaces at All Levels Are Defined and Put Under Formal Control

Critical Interfaces Are Designed Based on OS Considerations

Interface Documentation Forms Part of the Formally Controlled Configuration Baseline

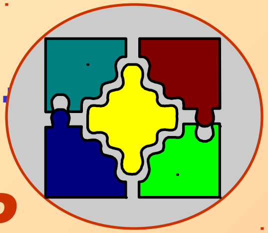
Interface Management Process - Planning

Characterize Interfaces

- Reference Models
- Building Codes/ Technical Architecture

Identify Opportunities:

- Rapidly Changing Tech
- Sub-system Likely to Grow or Evolve (Requirements Growth)
- High LC Cost Drivers
- Multiple Sources for Sub-system/components



Repeat Process With Each SEP

Interface Management Process - Design and Control

Assume Evolutionary Acquisition

Use Functional Partitioning and Modular Design to Develop Alternative Interface Designs

- Identify Critical and Opportunity Interfaces

- Create Firewalls

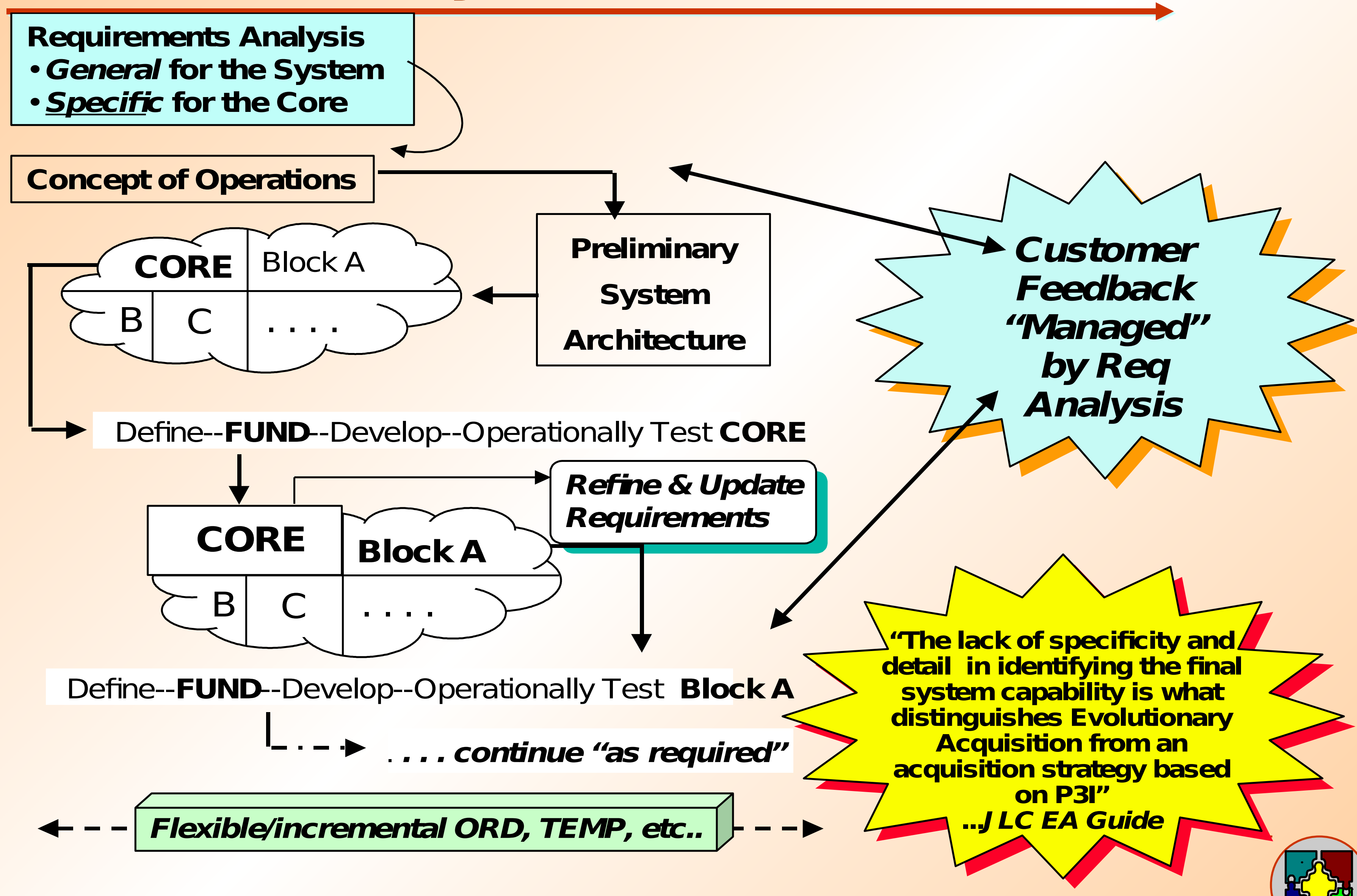
Use Trade Studies to Choose Interfaces

Define and Control Critical and Opportunity Interfaces

Track and Document Interface Definitions and Their Changes

Repeat Process With Each SEP

Evolutionary Acquisition



Modular Design

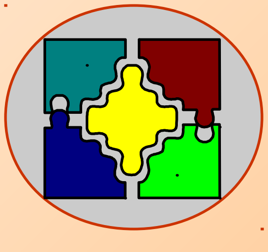
Develop Basic System Decomposition and Key Interfaces (More Depth As Design Proceeds)

Architecture Should Be Based On Functional Partitioning and Modular Design:

- One Module to Change Without Affecting Others
- Isolate Software From Hardware

Modular Design Means Compartmentalized Design,

And Does Not Necessarily Result in “Plug-n-Play” Style Modules.



Functional Partitioning & Modular Design Focus

Low Connectivity

Relationship Between Internal Elements of Different Modules

Creates Complex Interfaces

High Cohesion

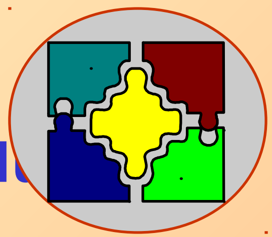
Similarity of Tasks Performed Within a Module

Use of Similar or Like Components (Series or Family)

One Component to Perform Multiple Tasks

Low Coupling

Measure of Interdependence Between Modu



High Level of Information Sharing

Priority of Interface Design Approaches

Open Standards That Allow Competitive Products,

Open Interface Design That Allows Installation of Competitive Products With Minimal Change,

Open Interface Design That Allows Minimal Change Installation of Commercial or NDI Products Currently or Planned to Be in DOD Use, and Last,

Unique Design With Interfaces Designed With Upgrade Issues Considered.

First Priority: Use Open Standards

Using Open Standards Allows Technology Refresh Through Normal Maintenance and Provides Configuration Control Without Controlling the Configuration

Standards Should Be Either:

- Publicly Available From Consensus-based Industry Standards Bodies (Preferred), or
- De Facto

Profile the Interfaces Based on Selection of Standards and Options in Standards

Interface Design Teams

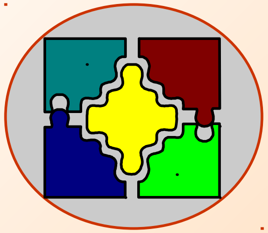
When Using an Open System Approach, Interfaces Are Designed, Not Just Identified

Interface Design Teams Are Integrated Teams (IPT) Representing Key Stakeholders

- **Interfacing Component Design Teams**
- **Management of Lifecycle Functions**
- **Specialists (Risk Management, Standards Selection, Modeling and Simulation, Etc.)**

Interface Design Teams

Duties Include:



- **Definition of Interfaces**
- **Lifecycle Planning**
- **Conformance Management Through Inter-team Action**
- **Interface and Conformance Documentation**

Interface Design Teams Develop Interface Documentation.

After Baseline Approval, Interfaces Should Be Formally Controlled by the Configuration Management Process

Conformance Management

Conformance Management Is Required to Verify That Interfaces Exactly Conform to the Standard

Conformance Management Ensures That Components From Multiple Suppliers Conform at the Interface Level

No Proprietary Implementations

OUTPUTS: SYSTEM ARCHITECTURE AND OPEN SYSTEMS

A SYSTEM’S ARCHITECTURE IS DEFINED BY ITS PATTERN OF INTERFACES

AT EACH INTERFACE THERE WILL BE SOME RANGE OF VALUES OF FORM, FIT, AND FUNCTION, WHICH WHEN ACHIEVED PROVIDES THE REQUIRED SYSTEM PERFORMANCE

OPEN SYSTEMS HAVE SPECIFIED AND ARRANGED INTERFACES TO PERMIT EASY CHANGE

Process Outputs

- Conformance Requirements Flow Down Through Baselines:

Sys Spec > CI Performance Spec > CI Detail Spec > TDP

- System Architectures Include Interface Definitions

- Data Base Includes Interface Management Trade Studies

Critical Item, Choice of Interface Standards, Choice of Interface Locations (Firewalls)

The Final System Design Will Usually Include Some Items That Are “Open” and Some That Are Not -- It Is Neither Necessary nor Often Possible That Every Element of a Complex System Be Totally Open.

Lifecycle Functions

Time and Cost to Upgrade a System Is Reduced.

Use of Competitive Products to Support the System

Conformance Management Is a Lifecycle Process

Lifecycle Integration

Use Integrated Teams To:

- Coordinate Government-Contractor Activities
- Design and Control System Interfaces
- Incorporate Lifecycle Considerations
- Develop and Control Architectures

Summary

Open Systems Is an Enabler of Systems Engineering Management

Open Systems Emphasizes Interface Management, Functional Participation, Modular Design, Multiple Design Solutions, and Conformance Management

It Emphasizes Use of Competitive Commercial Components Based on Interface Requirements Defined by Consensus Standards.

BACKUP SLIDES

Definition of Open Systems

"... A system that implements sufficient open specifications for interfaces, ...

... to enable properly engineered components to be utilized across a wide range of systems with minimal changes, ...

...to choose specifications and standards

- ☞ adopted by industry standards bodies *or*
- ☞ *de facto* standards (set by the market place)

for selected systems interfaces (functional and physical), products, practices, and tools."

DOD 5000.2-R

23 March 1996

AN OPEN SYSTEM IS A SYSTEM WHICH …

- IMPLEMENTS SUFFICIENT OPEN SPECIFICATIONS FOR INTERFACES, SERVICES, AND SUPPORTING FORMATS TO ENABLE PROPERLY ENGINEERED COMPONENTS TO BE UTILIZED ACROSS A WIDE RANGE OF SYSTEMS WITH MINIMAL CHANGES, AND

- CAN INTEROPERATE WITH OTHER COMPONENTS ON LOCAL AND REMOTE SYSTEMS IN A STYLE WHICH FACILITATES PORTABILITY

The Use Of "Building Codes"

Reference Model -

A logical representation of the system decomposition that clearly depicts key interfaces.

Open Standards -

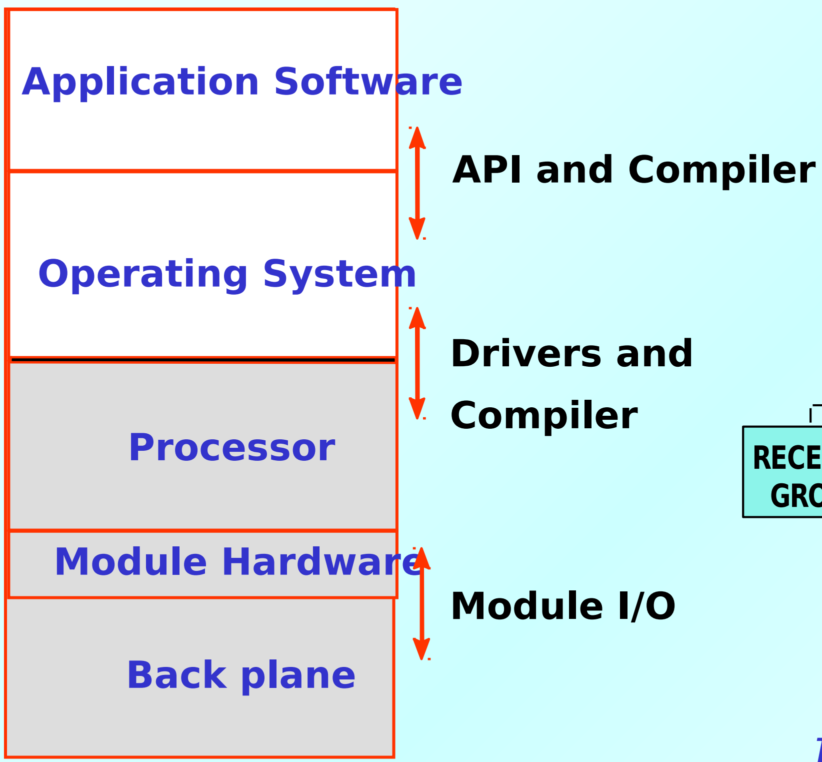
Publicly available, well defined consensus-based (formal, informal (market)) standards and practices.

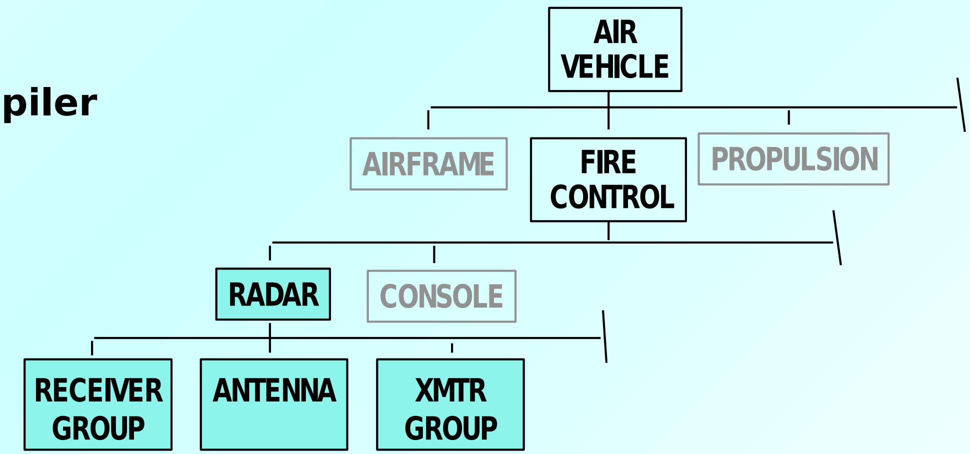
Output of System Description Design Phase

- Due by SFR

- Updated as Required

Reference Models





Simple Physical Decomposition (WBS)

Simple Electronics Reference Model